Arbeidsnotater

STATISTISK SENTRALBYRÅ
Dronningensgt. 16, Oslo-Dep., Oslo 1. Tlf. 41 38 20, 41 36 60

IO 73/5 31. februar 1973

METODEHEFTE NR. 5

Notater om to maskinprogrammer, om inntekt og forbruk, og om Hadwiger-funksjonen

Innhold

Forord

Metodehefter i serien Arbeidsnotater

I tilknytning til mange prosjekter i Statistisk Sentralbyrå utarbeides det mindre, upretensiøse notater for avklaring av spørsmål av metodisk interesse. Det kan dreie seg om utvalgsteknikk, alternative spørsmålsformuleringer, presentasjonsmetoder, begrepsavklaringer, diskusjon av "funn" i data, systemidéer, eller andre temaer. Selv om mange slike notater bare har begrenset interesse i ettertid, vil det blant dem være noen som kunne fortjene å bli mer alminnelig tilgjengelig enn de har vært hittil. Det kan også være nyttig å ha dem registrert sentralt, slik at det blir lettere å få oversikt over det stoffet som foreligger, og å referere tilbake til det.

Byrået har innført en publiseringsordning for stoff av dett slaget. Etter forbilde av serien Technical Notes fra U.S. Bureau of the Census publiserer en leilighetsvis et passende antall slike notater samlet i metodehefter i serien Arbeidsnotater. Inneværende hefte er det femte av denne typen.

Forsker Jan M. Hoem er oppnevnt som redaktør av metodeheftene. Medarbeidere i Byrået som lager stoff som kan være aktuelt, bes sende dette til redaksjonen etter hvert som det blir ferdig.

Kontorlederne bes holde øynene åpne for denne nye publiseringsmuligheten.

Assistent Liv Hansen er redaksjonssekretær.

SHØ/WA, 2/1-73

System description of the regional population projection programme of the Central Bureau of Statistics of Norway

by Synnøve Høvset

C o n t e n t s :

1. Introduction

Mathmatical aspects and a verbal description of the regional population projection model of the Central Bureau of Statistics of Norway has been given by Gilje (1969) and Nordbotten (1970). In the present note, some technical details will be given of the projection programme in its most recent version. Concerning superscripts and subscripts used, refer to appendix 3, section II.

This note is organized as follows.

In Section 2, a general description of the input/output system is given. The Section should be read in conjunction with the file descriptions, appended to this note.

In Section 3, a general programme description is given.

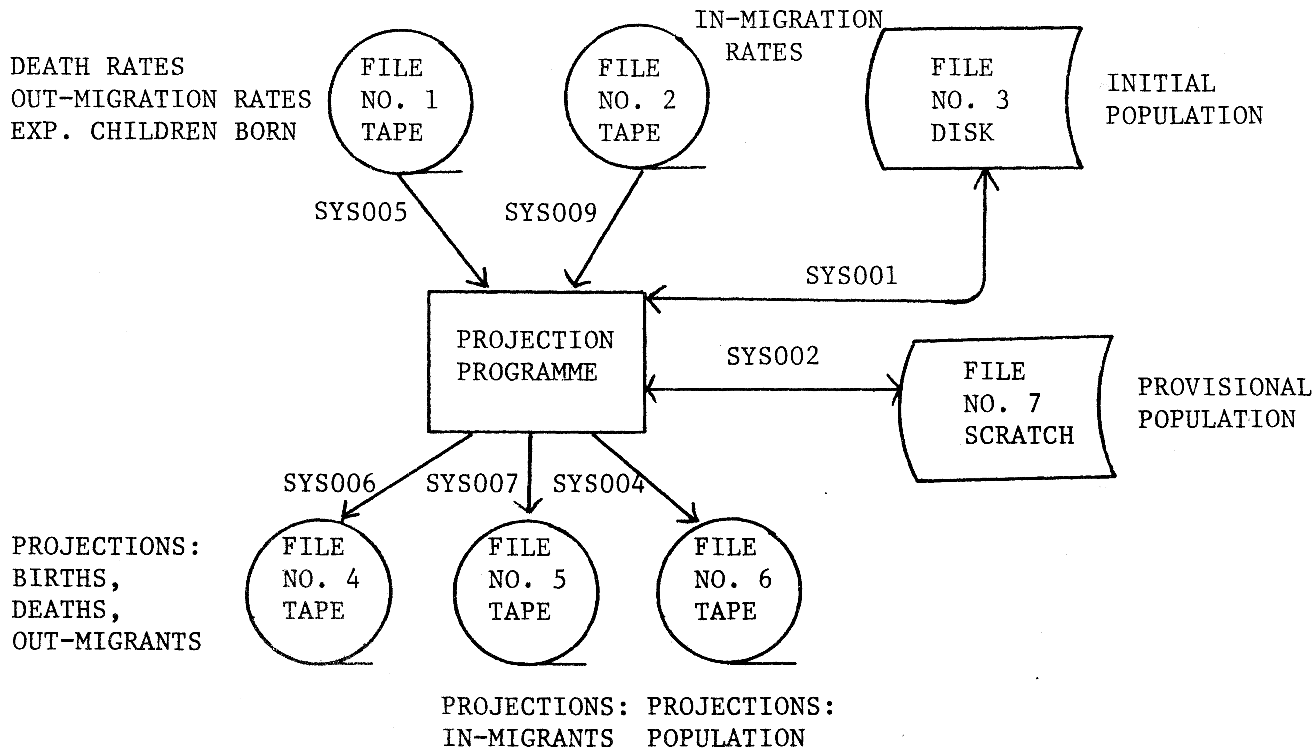
In Section 4, each section of the programme is described, and an explanation is given of the central variables of each section.

References

Gilje, Eivind (1969): "Model for population projections for Norwegian regions", Yearbook of Population Research in Finland, 1969, Vol. 11, pp. 22-32.

Gilje, Eivind and Svein Nordbotten (1970): "A demographic model for the Norwegian population and its technical characteristics", Statistical Review of the Swedish National Central Bureau of Statistics, III 9 (1): 13-24.

2. INPUT/OUTPUT



Explanation of some codes used for sex and age, and of data in the files:

Sex has code 1 for males, code 2 for females.

Age on the population file for the year n is computed as per December 31. Thus age x = 0 corresponds to children born in year n.

Age on the other files for the year n is computed as per January 1. Thus age x = 0 corresponds to persons born in year n-1. Age x = -1 corresponds to children born during projection year n.

Migration rates are set equal zero for ages greater than 69.

The projection years have codes 01, 02, ..., corresponding to years n+1, n+2, ..., where year n is the starting year from which we have the initial population.

On each file, there is one record for each municipality.

File no. 1. Death rates, out-migration rates, expected numbers of children born.

Death rates are given in 100 x 2 matrices organized by age (= -1, 0, ..., 98) and sex (= 1, 2). The rates are computed for the whole country, so in fact, the same matrix is used for all municipalities .

Out-migration rates are given in 71 x 2 matrices organized by age (= -1, 0, ..., 69) and sex (= 1, 2).

Expected numbers of children born are given in 30 x 1 matrices organized by mother's age (= 15, 16, ..., 44) and sex (= 2).

File no. 2. In-migration rates.

In-migration rates are given in 71 x 2 matrices organized by age (= -1, 0, ..., 69) and sex (= 1, 2).

File no.3. Initial population as per December 31, year n.

The population is given in 100 x 2 matrices organized by age (= 0, 1, ..., 99) and sex (= 1, 2). This is a combined input/output file. First the data are taken as an input to the computations. When the population for a projection year is computed, the result is returned as an output on this file. For the subsequent projection year, the data on the file is an input again, and so on.

File no. 4. Projections: Deaths, out-migrants, children born.

Total number of children born during the projection year, by sex (= 1, 2).

Numbers of children born during the projection year are given in a 30 x 1 matrix organized by mother's age (= 15, 16, ..., 44) and sex (= 1, 2).

Numbers of children born and dead during the projection year are given.

Numbers of deaths during the projection year are given in a 99 x 2 matrix organized by age (= 0, 1, ..., 98) and sex (= 1, 2).

Numbers of children born among out-migrants during the projection year are given.

Numbers of out-migrants during the projection year are given in a 70 x 2 matrix organized by age (= 0, 1, ..., 69) and sex (= 1, 2).

File no. 5. Projections: In-migrants

Numbers of children born among in-migrants during the projection year are given.

Numbers of in-migrants are given in a 70 x 2 matrix organized by age (= 0, 1, ..., 70) and sex (= 1, 2).

File no. 6. Projections: Population.

The projected population is given in a 100 x 2 matrix organized by age (= 0, 1, ..., 99) and sex (= 1, 2). The starting population is not included on the file.

File no. 7 is used as a scratch file for a quantity called the provisional population.

The provisional population is given in a 100 x 2 matrix organized by age (=0, 1, ..., 99) and sex (= 1, 2). This population includes in-migrants, but not deaths and out-migrants. The file is a combined input/output file. It is first used for some output. Subsequently the data are used as an input again.

3. PROJECTION PROGRAMME DESCRIPTION

The programme system consists of a main programme written in FORTRAN, and 10 subroutines written in COBOL and FORTRAN. COBOL is used for READ/WRITE, FORTRAN for computations. The program can be run with a varying number of projection years, depending on LL and L.

LL = no. of first projection year in the current run.

L = no. of last projection year in the current run.

K = number of municipalities in the country.

CG = proportion of boys among births.

CP = proportion of girls among births.

The constant SWITCH has 3 values, and controls the subroutines RES1 and RES2. Each of them is called on three different occasions, and they carry out different operations each time, depending on the value of SWITCH.

When a new projection year and a new municipality starts, SWITCH is set egual to 0.

When the routines DATA, BIRTHS, DEATHS, MOVES, POPN1 and RES1 have been used for each municipality within one projection year, SWITCH is set equal to 2.

When all projection years and all municipalities are finished, SWITCH is set equal to 1.

The new population computed for projection year n+1 is placed in the same area as the initial population for year n used in the computations. The new population is then used as the initial population for the next run.

The data for the new population come as output on file no. 3.

COBOL subroutines and files used:

DATA uses the following files as <u>INPUT</u>:

Initial population, file no. 3	BEDISK	on SYS001
Death and out-migration rates, expected children born, file no. 1	KOFILE	on SYS005

RES1 use the following files as <u>OUTPUT</u>:

Provisional population, file no. 7	FORBEST	on SYS002
Death and out-migration rates, expected children born, file no. 4	BEVEGELSE	on SYS006

DATA uses the following files as <u>INPUT</u>:

Provisional population, file no. 7	FORBEST	on SYS002
In-migration rates, file no. 2	INNFLYTT	on SYS009

RES2 uses the following files as <u>OUTPUT</u>:

Population on tape, file no. 6	PROGRES	on SYS004
Population on disk, file no. 3	BEDISK	on SYS001
In-migrants, file no. 5	BEVEGELSE	on SYS007

When the run starts, SWITCH is set equal to 0. Three DO-loops, A, B and C, are described in Section 4 below. Loop C is performed once for each projection year, I = LL, LL+1, ..., L. Loop A is performed for all municipalities within each projection year, J = 1, 2, ..., K. Loop B is performed for all municipalities within each projection year, J = 1, 2, ..., K.

4. SUBROUTINE DESCRIPTION

C1.

A1. CALL DATA (COBOL)

Read file no. 1, KOFILE and file no. 3, BEDISK. Put data into different variables.

Open files before first municipality and new projection year.

Close files after first municipality and new projection year.

NR = municipality number from file no. 3.

NUMBER corresponds to the number of municipalities, K.

F = Expected no. of children born, from file no. 1.

Q = Death rates, from file no. 1.

UT = Out-migration rates, from file no. 1.

BG = Population, from file no. 3.

A2. CALL BIRTHS (FORTRAN)

The variables F and BG come from DATA. The variables FF, A, NG and NP are computed.

F = expected no. of children.

BG = population.

FF = no. of children born.

A = total no. of children born.

NG = no. of boys born during the projection year.

NP = no. of girls born during the projection year.

The formulas used:

$$\left.\begin{aligned} FF_x^k &= F_x^k \cdot {}^{F}BG_x^k \\ A^k &= \sum_x FF_x^k \end{aligned}\right\} \text{, age } x = 15, 16, \ldots, 44,$$

$$NG^k = A^k \cdot CG,$$

$$NP^k = A^k \cdot CP.$$

A3. CALL DEATHS (FORTRAN)

The variables BG and Q come from DATA. NG and NP come from BIRTHS. The variable D is computed.

BG = population.

Q = death rates (computed for the whole country, used for each municipality).

NG/NP = no. of boys/girls born during the projection year.

D = no. of deaths.

The formulas used:

$$^{1}D_{-1}^{k} = NG^{k} \cdot Q_{-1}^{k}, \quad \text{age } x = -1, \text{ sex} = 1,$$

$$^{2}D_{-1}^{k} = NP^{k} \cdot Q_{-1}^{k}, \quad \text{age } x = -1, \text{ sex} = 2,$$

$$D_{x} = BG_{x}^{k} \cdot Q_{x}^{k}, \quad \text{age } x = 0, 1, \ldots, 98.$$

A4. CALL MOVEO (FORTRAN)

The variables BG and UT come from DATA. NG and NP come from BIRTHS. UA is defined in COMMON-statement. The variable U is computed.

BG = population.

UT = out-migration rates.

NG/NP = no. of boys/girls born during the projection year.

UA = no. of out-migrants for the whole country.

U = no. of out-migrants in one municipality.

The formulas used:

$$^{1}U_{-1}^{k} = NG^{k} \cdot UT_{-1}^{k}, \quad \text{age } x = -1, \text{ sex} = 1,$$

$$^{2}U_{-1}^{k} = NP^{k} \cdot UT_{-1}^{k}, \quad \text{age } x = -1, \text{ sex} = 2,$$

$$U_{x}^{k} = BG_{x}^{k} \cdot UT_{x}^{k}, \quad \text{age } x = 0, 1, \ldots, 69,$$

$$UA_{x} = \sum_{k} U_{x}^{k}, \quad \text{age } x = -1, 0, 1, 2, \ldots, 69$$

A5. CALL POPN1 (FORTRAN)

The variable BG comes from DATA. U comes from MOVEO. D comes from DEATH. NG and NP come from BIRTHS. The variable BNF is computed.

BG = population.

U = no. of out-migrants.

D = no. of deaths.

NG/NP = no. of boys/girls born during the projection year.

BNF = provisional population.

The formulas used:

$$^{1}BNF_{0}^{k} = NG^{k} - U_{-1}^{k} - D_{-1}^{k}, \quad \text{age } x = -1, \text{ sex} = 1,$$

$$^{2}BNF_{0}^{k} = NP^{k} - U_{-1}^{k} - D_{-1}^{k}, \quad \text{age } x = -1, \text{ sex} = 2,$$

$$BNF_{x+1}^{k} = BG_{x}^{k} - D_{x}^{k} - U_{x}^{k}, \quad \text{age } x = 0, 1, \ldots, 69,$$

$$BNF_{x+1}^{k} = BG_{x}^{k} - D_{x}^{k}, \quad \text{age } x = 70, 71, \ldots, 98.$$

A5. CALL RES1 (COBOL)

At this point, SWITCH will always be equal to 0, and the routine is used for WRITE.

Write data on file no. 7, FORBEST, and file no. 4, BEVEGELSE.

Open output file no. 4 at the first projection year and first municipality. Open output file no. 7 at new projection year and first municipality.

The variable NR comes from DATA. NG, NP and FF come from BIRTHS. D comes from DEATHS. U comes from MOVEO. BNF comes from POPN1. SWITCH and J come from the main programme.

NR = municipality no.

NG/NP = no. of boys/girls born during the projection year, on output file no. 4.

FF = no. of children born on mothers age, on output file no. 4.

D = no. of deaths, on output file no. 4.

U = no. of out-migrants, on output file no. 4.

BNF = provisional population, on output file no. 7.

End of loop A.

C2. SWITCH = 2.

C3. CALL RES1 (COBOL).

At this point, SWITCH is always equal to 2, and the routine is used only for control on OPEN/CLOSE. Close output file no. 7, FORBEST. At next call on RES1, we open this file again.

C4.

B1. CALL DATA2 (COBOL).

Read file no. 7, FORBEST, and file no. 2, INNFLYTT.

At new projection year and first municipality, input file no. 7 and input file no. 2 are opened. When all municipalities are through, both files are closed.

IN = in-migration rates, from input file no. 2.

BNF = provisional population, from input file no. 7.

B2. CALL MOVEIN (FORTRAN).

The variable IN comes from DATA2. UA comes from MOVEO. The variable INF is computed.

IN = in-migration rates.

UA = no. of out-migrants (for the whole country).

INF = no. of in-migrants.

Formula used: $INF_x^k = UA_x \cdot IN_x^k$, age x = -1, 0, 1, ..., 69.

B3. CALL POPN (FORTRAN).

The variable BNF comes from POPN1. INF comes from MOVEIN. The variable BN is computed.

BNF = provisional population.

INF = no. of in-migrants.

BN = population.

The formulas used: $BN_{x+1} = BNF_{x+1} + INF_x$, age x = -1, 0, ..., 69,

$BN_{x+1} = BNF_{x+1}$, age x = 70, 71, ..., 99.

B4. CALL RES2 (COBOL).

At this point, SWITCH is always equal to 2, and the routine is used for WRITE.

Write data on tapefile no. 6, PROGRES, on diskfile no. 3, BEDISK, and on file no. 5, BEVEGELSE.

Open output file no. 6 and output file no. 5 at first municipality and first projection year.

Open output file no. 3 at first municipality and new projection year.

The variable NR comes from DATA, BN and INF come from POPN. SWITCH and I come from the main programme.

NR = municipality no.

BN = population put on tapefile no. 6 and diskfile no. 3.

INF = no. of in-migrants on file no. 5.

The end of loop B.

C5. SWITCH = 0.

C6. CALL RES2 (COBOL).

At this point, SWITCH is always equal to 0, and the routine is only used for control on OPEN/CLOSE. Close output file no. 3, BEDISK. At next call on RES2, only this file will be opened.

The end of loop C.

SWITCH = 1.

CALL-RES1.

Here, SWITCH is always equal to 1, and the routine is used for CLOSE. Close output file no. 4, BEVEGELSE, and output file no. 7, FORBEST.

CALL RES2.

At this point, SWITCH is always equal to 1, and the routine is used for CLOSE. Close output files no. 3, BEDISK, no. 6, PROGRES, and no. 3, BEVEGELSE.

APPENDIX 1

FILE DESCRIPTION NO 1

File name: DEATH RATES, OUT-MIGRATION RATES, EXPECTED NUMBERS OF CHILDREN BORN.

File contents: Expected numbers of children born, in 30 x 1 matrix.
Death rates in 100 x 2 matrix.
Out-migration rates in 71 x 2 matrix.
1 record per municipality.

Field no.	Pos. From-To	Data description	Field description
1	1-4	EBC	Municipality
2	5-8	"	Filler
3	9-12	FLOATING POINT	Exp. children born, mother's age 15
4	13-16	"	" " " " " 16
:	:	:	:
32	125-128	"	" " " " 30
33	129-132	"	DEATH RATES age -1, sex 1
34	133-136	"	" " age -1, sex 2
35	137-140	"	" " age 0, sex 1
:	:	:	:
232	925-928	"	" " age 98, sex 2
233	929-932	"	OUT-MIGRATION RATES, age -1, sex 1
			" " age -1, sex 2
:	:	:	" " age 0, sex 1
			:
374	1493-1496	Floating point	" " age 69, sex 2

SORT: pos. 1-4, DOS standard label, Blockfactor = 1

FILE DESCRIPTION NO 2

File name: IN-MIGRATION RATES

File contents: In-migration rates in 71 x 2 matrix.
1 record per municipality.

Field no.	Pos. From-To	Data description	Field description
1	1-4	EBC	Filler
2	5-8	"	Municipality
3	9-12	Floating point	In-migr. rates, age -1, sex 1
4	13-16	"	" " age -1, sex 2
5	17-20	"	" " age 0, sex 1
⋮	⋮	⋮	⋮
141	569-572	"	" " age 69, sex 1
142	573-576	Floating point	" age 69, sex 2

SORT: pos. 5-8, DOS standard label, Blockfactor = 1.

FILE DESCRIPTION NO 3

File name: INITIAL POPULATION as per December 31.

File contents: Numbers of persons alive as per Dec. 31 in the initial year, in a 100 x 2 matrix.
1 record per municipality.

Field no.	Pos. From-To	Data description	Field description
1	1-2	EBC	Filler
2	3-6	"	Municipality
3	7-8	"	Filler
4	9-12	Floating point	Population age 0, sex 1
5	13-16	"	" age 0, sex 2
6	17-20	"	" age 1, sex 1
⋮	⋮	⋮	⋮
203	805-808	"	" age 99, sex 2

SORT: pos. 3-6, DOS standard label, Blockfactor = 2.

FILE DESCRIPTION NO 4

File name: PROJECTIONS: DEATHS, OUT-MIGRANTS, CHILDREN BORN

File contents: Numbers of girls/boys born.

Numbers of deaths among boys/girls born during the projection year.

Numbers of deaths in 99 x 2 matrix.

Numbers of out-migrants in 70 x 2 matrix.

Numbers of children born in 30 x 1 matrix (by mother's age).

1 record per year per municipality.

Field no.	Pos. From-To	Data description	Field description
1	1-2	EBC	Projection year (01, 02, ...)
2	3-6	"	Municipality
3	7-8	"	Filler
4	9-12	Floating point	Girls born in the projection year
5	13-16	"	Boys born in the projection year
6	17-20	"	Deaths among boys born in the projection year
7	21-24	"	Deaths among girls born in the projection year
8	25-816	"	Deaths, age (= 0, 1, ..., 98) x sex (= 1, 2)
9	817-820	"	Out-migrants, boys born in the projection year
10	821-824	"	Out-migrants, girls born in the projection year
11	825-1384	"	Out-migrants age (= 0, 1, ..., 69) x sex (= 1, 2)
12	1385-1504	Floating point	Children born by mother's age (= 15, 16, ..44)

SORT: pos. (1-2) x (3-6), STANDARD DOS LABEL, BLOCKFACTOR = 1.

FILE DESCRIPTION NO 5

File name: PROJECTIONS: IN-MIGRANTS

File contents: Numbers of in-migrants in 71 x 2 matrix.
1 record per year per municipality.

Field no.	Pos. From-To	Data description	Field description
1	1-2	EBC	Projection year (01, 02, ...)
2	3-4	"	Filler
3	5-8	"	Municipality
4	9-12	FLOATING POINT	In-migrants, age -1, sex 1
5	13-16	"	In-migrants, age -1, sex 2
6	17-20	"	In-migrants, age 0, sex 1
7	21-24	"	In-migrants, age 0, sex 2
⋮	⋮	⋮	⋮
75	573-576	FLOATING POINT	In-migrants, age 69, sex 2

SORT: pos. (1-2) x (5-8), STANDARD DOS LABEL, BLOCKFACTOR = 1.

FILE DESCRIPTION NO 6

File name: PROJECTIONS: POPULATION

File contents: Numbers of persons alive in 100 x 2 matrix.
1 record per year per municipality.

Field no.	Pos. From-To	Data description	Field description
1	1-2	EBC	Projection year (01, 02, ...)
2	3-6	"	Municipality
3	7-8	"	Filler
4	9-12	FLOATING POINT	Number of persons age 0 sex 1
5	13-16	"	Number of persons age 0 sex 2
6	17-20	"	Number of persons age 1 sex 1
⋮	⋮	⋮	⋮
203	805-808	FLOATING POINT	Number of persons age 99 sex 2

SORT: Pos. (1-2) x (3-6), STANDARD DOS LABEL, BLOCKFACTOR = 1.

APPENDIX 2

DOS FORTRAN IV 360N-FO-479 3-6 MAINPGM DATE 15/02/73 TIME

```
      DIMENSION F(30),BG(2,100),Q(2,100),UT(2,71),U(2,71),BNF(2,101),
     1D(2,100),BN(2,100),IN(2,71),INF(2,71),FF(30)
      COMMON A, CG, CP, NUG, NUP, UA(2,71),J,L,M,N
      REAL NG,NP,NUG,NUP,INF,IN
      INTEGER SWITCH
      LL = 25
      L = 29
      K = 444
      CG = 0.51543
      CP = 0.48457
      SWITCH = 0
      DO 1 I=LL,L
      DO 2 J=1,K
      CALL DATA(NR, F, Q, UT, BG,K)
      CALL BIRTHS (F,BG,NG,NP,FF)
      CALL DEATHS (BG,Q,D,NG,NP)
      CALL MOVEO (BG,UT,U,NG,NP)
      CALL POPN1 (BG,BNF,U,D,NG,NP)
      CALL RES1 (NR,NG,NP,BNF,D,U,SWITCH,I,FF)
    2 CONTINUE
      SWITCH = 2
      CALL RES1 (NR,NG,NP,BNF,D,U,SWITCH,I,FF)
      DO 3 J=1,K
      CALL DATA2 (NR,BNF,IN, K)
      CALL MOVEIN (INF,IN)
      CALL POPN (BN,BNF,INF)
    3 CALL RES2 (NR,BN,INF,SWITCH,I)
      SWITCH =0
      CALL RES2 (NR,BN,INF,SWITCH,I)
    1 CONTINUE
      SWITCH = 1
      CALL RES1 (NR,NG,NP,BNF,D,U,SWITCH,I,FF)
      CALL RES2 (NR,BN,INF,SWITCH,I)
      END
```

DOS FORTRAN IV 360N-FO-479 3-6 BIRTHS DATE 15/02/73 TIME

```
      SUBROUTINE BIRTHS (F,BG,NG,NP,FF)
      DIMENSION F(30),BG(2,100),FF(30)
      REAL NG,NP
      COMMON A, CG, CP, NUG, NUP, UA(2,71),J,L,M,N
      A=0.0
      DO 100 M=1,30
      FF(M) = F(M)*BG(2,M+14)
  100 A=A+FF(M)
      NG=A*CG
      NP=A*CP
      RETURN
      END
```

DOS FORTRAN IV 360N-FO-479 3-6 DEATHS DATE 15/02/73 TIME

```
      SUBROUTINE DEATHS (BG,Q,D,NG,NP)
      DIMENSION BG(2,100),Q(2,100),D(2,100)
      COMMON A, CG, CP, NUG, NUP, UA(2,71),J,L,M,N
      REAL NP, NG
      D(1,1) = NG*Q(1,1)
      D(2,1) = NP*Q(2,1)
      DO 200 M=1,2
      DO 200 N=1,99
  200 D(M,N+1) = BG(M,N)*Q(M,N+1)
      RETURN
      END
```

DOS FORTRAN IV 360N-FO-479 3-6 MOVEO DATE 15/02/73 TIME

```
      SUBROUTINE MOVEO (BG,UT,U,NG,NP)
      DIMENSION BG(2,100),UT(2,71),U(2,71)
      REAL NUG, NUP, NG, NP
      COMMON A, CG, CP, NUG, NUP, UA(2,71),J,L,M,N
      U(1,1) = NG*UT(1,1)
      U(2,1) = NP*UT(2,1)
      DO 700 M=1,2
      DO 700 N=1,70
  700 U(M,N+1) = BG(M,N)*UT(M,N+1)
      IF(J-1) 302,302,303
  302 DO 305 M=1,2
      DO 305 N=1,71
  305 UA(M,N)=U(M,N)
      GO TO 311
  303 DO 304 M=1,2
      DO 304 N=1,71
  304 UA(M,N)=UA(M,N)+U(M,N)
  311 RETURN
      END
```

DOS FORTRAN IV 360N-FO-479 3-6 POPN1 DATE 15/02/73 TIME

```
      SUBROUTINE POPN1 (BG,BNF,U,D,NG,NP)
      DIMENSION BG(2,100),BNF(2,101),U(2,71),D(2,100)
      REAL NG,NP
      COMMON A, CG, CP, NUG, NUP, UA(2,71),J,L,M,N
      BNF(1,1) = NG-U(1,1)-D(1,1)
      BNF(2,1) = NP-U(2,1)-D(2,1)
      DO 400 M=1,2
      DO 400 N=1,70
  400 BNF(M,N+1) = BG(M,N)-D(M,N+1)-U(M,N+1)
      DO 402 M=1,2
      DO 402 N=1,29
  402 BNF(M,N+71) = BG(M,N+70)-D(M,N+71)
      RETURN
      END
```

DOS FORTRAN IV 360N-FO-479 3-6 MOVEIN DATE 15/02/73 TIME

```
      SUBROUTINE MOVEIN (INF,IN)
      REAL INF,IN
      DIMENSION IN(2,71),INF(2,71)
      COMMON A, CG, CP, NUG, NUP, UA(2,71),J,L,M,N
      DO 500 M=1,2
      DO 500 N=1,71
  500 INF(M,N)=UA(M,N)*IN(M,N)
      RETURN
      END
```

DOS FORTRAN IV 360N-FO-479 3-6 POPN DATE 15/02/73 TIME

```
      SUBROUTINE POPN (BN,BNF,INF)
      REAL INF
      DIMENSION BNF(2,101),BN(2,100),INF(2,71)
      COMMON A, CG, CP, NUG, NUP, UA(2,71),J,L,M,N
      DO 600 M=1,2
      DO 600 N=1,71
  600 BN(M,N) = BNF(M,N)+INF(M,N)
      DO 601 M=1,2
      DO 601 N=1,29
  601 BN(M,N+71) = BNF(M,N+71)
      RETURN
      END
```

DATA

```
LINE NO. SEQ. NO.          SOURCE STATEMENT                                    CBD CL

     00100  IDENTIFICATION DIVISION.
     00105  PROGRAM-ID. 'CODATA'.
     00110  ENVIRONMENT DIVISION.
     00115  INPUT-OUTPUT SECTION.
     00120  FILE-CONTROL.
     00125      SELECT BEDISK ASSIGN TO 'SYS001' UTILITY 2314.
     00130      SELECT KOFILE ASSIGN TO 'SYS005' UTILITY 2400.
            DATA DIVISION.
            FILE SECTION.
     00135  FD  BEDISK, DATA RECORDS ARE DRECORD, LABEL RECORDS ARE
     00140      STANDARD, RECORDING MODE IS F, BLOCK CONTAINS 3 RECORDS.
     00145  01  DRECORD.
     00146      02 FILLER, PICTURE IS XX.
     00150      02 MUNNR, PICTURE IS 9(4).
                02  FILLER PICTURE XX.
     00160      02 MATRISE.
     00165        03 KJONN, OCCURS 100 TIMES.
     00170          04 ALDER, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
     00175  FD  KOFILE, DATA RECORDS ARE KORECORD, LABEL RECORDS ARE
     00180      STANDARD, RECORDING MODE IS F.
     00185  01  KORECORD.
     00190      02 KOMNR, PICTURE 9(4).
     00191      02 FILLER                    PICTURE XXXX.
     00195      02 KOFOD.
     001951       03 ALDFO, OCCURS 30 TIMES, USAGE IS COMPUTATIONAL-1.
     00200  02 KODOD.
     00205        03 AGE, OCCURS 100 TIMES.
     00210          04 SEX, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
     00215      02 KOFLYT.
     00220        03 AG, OCCURS 71  TIMES.
     00225          04 SE, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
     002301 WORKING-STORAGE SECTION.
     002302 77  L, PICTURE IS 9(10), VALUE IS 0.
     002303 77  K, PICTURE IS 9(10), VALUE IS 0.
     00231  77  M, PICTURE 9(5), USAGE IS COMPUTATIONAL.
     00235  LINKAGE SECTION.
     00240  77  NR, PICTURE 9(5), USAGE IS COMPUTATICNAL.
     00241  77  NUMBER          PICTURE 9(5), USAGE IS COMPUTATIONAL.
     00245  01  FA.
S    002001     02  F, OCCURS 30 TIMES,
     002051             USAGE IS COMPUTATIONAL-1.
     00250  01  QB.
S    002101     02 QA, OCCURS 100 TIMES.
     002151       03  Q, OCCURS 2 TIMES,
     002201             USAGE IS COMPUTATIONAL-1.
     00255  01  UB.
S    002251     02 UA, OCCURS 71  TIMES.
     002301       03 UT, OCCURS 2 TIMES,
     002351             USAGE IS COMPUTATIONAL-1.
     00260  01  BB.
S    002401     02 BA, OCCURS 100 TIMES.
     002451       03 BG, OCCURS 2 TIMES,
     002452             USAGE IS COMPUTATIONAL-1.
     00265  PROCEDURE DIVISION.
     00270      ENTER LINKAGE.
     00275      ENTRY 'DATA', USING NR,  F (1),   Q (1, 1), UT (1, 1),
     00276      BG (1, 1), NUMBER.
     00280      ENTER COBOL.
     00285  DATA.
     00290  A.  GO TO B.
     00295  B.  OPEN INPUT BEDISK, KOFILE.
     00300      ALTER A TO PROCEED TO C.
     00305  C.  READ BEDISK,  AT END GO TO SLUTT.
     00310      READ KOFILE,   AT END GO TO SLUTT.
            ALTER-ZERO-M. GO TO ZERO-M.
            ZERO-M. MOVE 0 TO M.
                ALTER ALTER-ZERO-M TO PROCEED TO DD.
     00311  DD. ADD 1 TO M.
     00315      MOVE MUNNR TO NR.
     00320      IF KOMNR IS NOT EQUAL TO NR GO TO ERROR.
     00325  D.  ADD 1 TO L.
     00330      MOVE ALDFO (L) TO  F (L).
     00335      IF L IS LESS THAN 30 GO TO D.
     00340      MOVE 0 TO L.
```

```
LINE NO. SEQ. NO.            SOURCE STATEMENT

         00345  E.  ADD 1 TO L.
         00350  FE. ADD 1 TO K.
         00355      MOVE SEX (L, K) TO  Q (L, K).
         00360      IF K IS LESS THAN 2 GO TO FE.
         00365      MOVE 0 TO K.
         00370      IF L IS LESS THAN 100 GO TO E.
         00375      MOVE 0 TO L.
         00380  EE. ADD 1 TO L.
         00385  FF. ADD 1 TO K.
                    MOVE SE (L, K) TO UT (L, K).
         00395      IF K IS LESS THAN 2 GO TO FF.
         00400      MOVE 0 TO K.
         00405      IF L IS LESS THAN 71  GO TO EE.
         00410      MOVE 0 TO L.
         00415  EEE. ADD 1 TO L.
         00420  FFF. ADD 1 TO K.
         00425      MOVE ALDER (L, K) TO BG (L, K).
         00430      IF K IS LESS THAN 2 GO TO FFF.
         00435      MOVE 0 TO K.
         00440      IF L IS LESS THAN 100 GO TO EEE.
         00445      MOVE 0 TO L.
         00446      IF M = NUMBER, GO TO SLUTT.
         00450      GO TO RETUR.
         00455  ERROR. STOP RUN.
         00460  SLUTT.
                    MOVE 0 TO M.
         00465      CLOSE BEDISK, KOFILE.
         00467      ALTER A TO PROCEED TO B.
         00470  RETUR.
         00475      ENTER LINKAGE.
         00480      RETURN.
         00485      ENTER COBOL.
```

RES1

```
LINE NO. SEQ. NO.            SOURCE STATEMENT                          CBD CL3-9

         00100  IDENTIFICATION DIVISION.
         00105  PROGRAM-ID. 'CORES1'.
         00110  REMARKS. SUBROUTINE SOM SKRIVER FOEDSLER, DOEDSFALL, UTFLYTTERE
         00115      OG FORELOEPIG BESTAND PÅ TAPE.
         00120  ENVIRONMENT DIVISION.
         00125  INPUT-OUTPUT SECTION.
         00130  FILE-CONTROL.
         00135      SELECT FORBEST ASSIGN TO 'SYS002' UTILITY 2314,
                        RESERVE NO ALTERNATE AREA.
         00140      SELECT BEVEGELSE ASSIGN TO 'SYS006' UTILITY 2400,
                        RESERVE NO ALTERNATE AREA.
         00145  DATA DIVISION.
         00150  FILE SECTION.
         00155  FD  FORBEST, DATA RECORDS ARE BESTRECORD1, LABEL RECORDS ARE
         00160      STANDARD, RECORDING MODE IS F.
         00165  01  BESTRECORD1.
         00166  02  FILLER                          PICTURE XX.
         00171      02 KOMMNR, PICTURE IS 9(4).
S        00166  02  FILLER                          PICTURE XX.
S        00166      02 SKRUE.
         00170      03 DONALD, OCCURS 100 TIMES.
         00175      04 DOLLY, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
         00180  FD  BEVEGELSE, DATA RECORDS ARE BEVRECORD, LABEL RECORDS ARE
         00185      STANDARD, RECORDING MODE IS F.
         00190  01  BEVRECORD.
                    02 PROGAR, PICTURE 99.
         00195      02 KOMNR, PICTURE IS 9(4).
                    02  FYLL1  PICTURE X.
         00200  02 FOEDSLER.
                    03  FYLL2  PICTURE X.
         00205      03 PIKEF, USAGE IS COMPUTATIONAL-1.
         00210      03 GUTTEF, USAGE IS COMPUTATIONAL-1.
         00215      02 DOEDSFALL.
         00220      03 DOE, OCCURS 100 TIMES.
         00225      04 DOED, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
         00230      02 UTFLYTTING.
         00235      03 UT, OCCURS 71  TIMES.
         00240      04 UTF, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
                    02  FOEDTE-E-MORS-ALDER.
                        03  FOEDTE-MORS-ALDER, OCCURS 30 TIMES,
```

```
LINE NO. SEQ. NO.          SOURCE STATEMENT

                                  USAGE IS COMPUTATIONAL-1.
         00245   WORKING-STORAGE SECTION.
         00250   77  L, PICTURE IS 9(5), VALUE IS 0.
         00255   77  K, PICTURE IS 9(5), VALUE IS 0.
         00260   LINKAGE SECTION.
         00261   77 SWITCH, PICTURE 9(5), USAGE IS COMPUTATIONAL.
         00262   77 J, PICTURE 9(5), USAGE IS COMPUTATIONAL.
         00265   77  NR, PICTURE 9(5), USAGE IS COMPUTATICNAL.
         00270   77  NG, USAGE IS COMPUTATIONAL-1.
         00275   77  NP, USAGE IS COMPUTATIONAL-1.
                 01  FOEDTE-E-M-A.
                     02  FF OCCURS 30 TIMES, USAGE IS COMPUTATIONAL-1.
         00280   01  BA.
         00285       02 BB, OCCURS 100 TIMES.
         00290       03 BNF, OCCURS 2 TIMES,
         00295       USAGE IS COMPUTATIONAL-1.
         00300   01  DO.
         00305       02 DOA, OCCURS 100 TIMES.
         00310       03 D, OCCURS 2 TIMES,
         00315       USAGE IS COMPUTATIONAL-1.
         00320   01  UA.
         00325       02 UB, OCCURS 71  TIMES.
         00330       03 U, OCCURS 2 TIMES,
         00335       USAGE IS COMPUTATIONAL-1.
         00340   PROCEDURE DIVISION.
         00345       ENTER LINKAGE.
         00350       ENTRY 'RES1', USING NR, NG, NP, BNF (1, 1), D (1, 1),
         00355       U (1, 1), SWITCH, J, FF (1).
         00360       ENTER COBOL.
         00365   RES1.
         00366   AAA. GO TO AAC.
         00367   AAC. OPEN OUTPUT BEVEGELSE.
         00375   AAB. OPEN OUTPUT FORBEST.
                     ALTER AAA TO PROCEED TO AC.
                 AC. IF SWITCH IS EQUAL TO 1 GO TO SLUTT,
         00384       ELSE IF SWITCH IS EQUAL TO 2 GO TO SLUTT1.
         00385   AAG. MOVE NR TO KOMNR.
         00390        MOVE NR TO KOMMNR.
                     MOVE J TO PROGAR.
         00395        MOVE NG TO GUTTEF.
         00400        MOVE NP TO PIKEF.
         00405   AD. ADD 1 TO K.
         00410   AE. ADD 1 TO L.
         00415       MOVE BNF (K, L) TO DOLLY (K, L).
         00420       MOVE D (K, L) TO DOED (K, L).
         00430       IF L IS LESS THAN 2 GO TO AE.
         00435       MOVE 0 TO L.
         00440       IF K IS LESS THAN 100 GO TO AD.
         00445       MOVE 0 TO K.
                     MOVE 0 TO L.
                 AD2.
                     ADD 1 TO K.
                 AE2.
                     ADD 1 TO L.
  S      00425       MOVE U (K, L) TO UTF (K, L).
                     IF L IS LESS THAN 2 GO TO AE2.
                     MOVE 0 TO L.
                     IF K IS LESS THAN 71 GO TO AD2.
                     MOVE 0 TO K.
                 AD3.
                     ADD 1 TO K.
                     MOVE FF (K) TO FOEDTE-MORS-ALDER (K).
                     IF K IS LESS THAN 30 GO TO AD3.
                     MOVE 0 TO K.
                     MOVE ZEROES TO FYLL1, FYLL2.
         00460       WRITE BESTRECORD1.
         00465       WRITE BEVRECORD.
                     GO TO RETUR.
         00466   SLUTT.
         00470       CLOSE FORBEST, BEVEGELSE.
         004711      GO TO RETUR.
  S      00471   SLUTT1.
         00472       CLOSE FORBEST.
         00474       ALTER AAA TO PROCEED TO AAB.
         00475   RETUR.
         00480       ENTER LINKAGE.
         00485       RETURN.
         00490       ENTER COBOL.
```

DATA2

```
LINE NO. SEQ. NO.          SOURCE STATEMENT                                CBD CL3-9

         00100  IDENTIFICATION DIVISION.
         00105  PROGRAM-ID. 'CODATA2'.
         00110  REMARKS. SUBROUTINE SOM LESER FORELOEPIG BESTAND OG
         00115       INNFLYTTINGSANDELER FOR EN KOMMUNE.
         00120  ENVIRONMENT DIVISION.
         00125  INPUT-OUTPUT SECTION.
         00130  FILE-CONTROL.
         00135       SELECT FORBEST ASSIGN TO 'SYS002' UTILITY 2314.
         00140       SELECT INNFLYTT ASSIGN TO 'SYS009' UTILITY 2400.
         00145  DATA DIVISION.
         00150  FILE SECTION.
         00155  FD   FORBEST, DATA RECORDS ARE BESTRECORD1, LABEL RECORDS ARE
         00160       STANDARD, RECORDING MODE IS F.
         00165  01   BESTRECORD1.
         00166  02   FILLER                           PICTURE X(2).
         00170       02 KOMMNR, PICTURE IS 9(4).
                02   FILLER                           PICTURE X(2).
S        00166       02 SKRUE.
         00175       03 DONALD, OCCURS 100 TIMES.
         00180       04 DOLLY, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
         00185  FD   INNFLYTT, DATA RECORDS ARE INNRECORD, LABEL RECORDS ARE
         00190       STANDARD, RECORDING MODE IS F.
         00195  01   INNRECORD.
         00196       02  YEAR                         PICTURE X(4).
         00200       02 KOMNR, PICTURE 9(4).
         00205       02 IFL.
         00210       03 IFLA, OCCURS 71  TIMES.
         00215       04 IFLB, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
         00220  WORKING-STORAGE SECTION.
         00225  77   L, PICTURE IS 9(5), VALUE IS 0.
         00230  77   K, PICTURE IS 9(5), VALUE IS 0.
         00231  77   M, PICTURE 9(5), USAGE IS COMPUTATIONAL.
         00235  LINKAGE SECTION.
         00240  77   NR, PICTURE 9(5), USAGE IS COMPUTATIONAL.
         00241  77   NUMBER          PICTURE 9(5), USAGE IS COMPUTATIONAL.
         00245  01   BA.
         00250       02 BB, OCCURS 100 TIMES.
         00255       03 BNF, OCCURS 2 TIMES,
         00260       USAGE IS COMPUTATIONAL-1.
         00265  01   IA.
         00270       02 IB, OCCURS 71  TIMES.
         00275       03 IN, OCCURS 2 TIMES,
         00280       USAGE IS COMPUTATIONAL-1.
         00285  PROCEDURE DIVISION.
         00290       ENTER LINKAGE.
         00295       ENTRY 'DATA2', USING NR, BNF (1, 1), IN (1, 1), NUMBER.
         00300       ENTER COBOL.
         00305  DATA2.
         00310  A.   GO TO B.
         00315  B.   OPEN INPUT INNFLYTT, FORBEST.
         00320       ALTER A TO PROCEED TO C.
         00325  C.   READ INNFLYTT, AT END GO TO SLUTT.
         00330       READ FORBEST, AT END GO TO SLUTT.
                ALTER-ZERO-M. GO TO ZERO-M.
                ZERO-M. MOVE 0 TO M.
                     ALTER ALTER-ZERO-M TO PROCEED TO DD.
         00331  DD.  ADD 1 TO M.
         00335       MOVE KOMMNR TO NR.
         00340       IF KOMNR IS NOT EQUAL TO NR GO TO ERROR.
         00345  D.   ADD 1 TO L.
         00350  D1.  ADD 1 TO K.
         00355       MOVE DOLLY (L, K) TO BNF (L, K).
         00360       IF K IS LESS THAN 2 GO TO D1.
         00365       MOVE 0 TO K.
         00370       IF L IS LESS THAN 100 GO TO D.
         00375       MOVE 0 TO L.
         00380  E.   ADD 1 TO L.
         00385  E1.  ADD 1 TO K.
         00390       MOVE IFLB (L, K) TO IN (L, K).
```

```
LINE NO. SEQ. NO.          SOURCE STATEMENT

         00395        IF K IS LESS THAN 2 GO TO E1.
         00400        MOVE 0 TO K.
         00405        IF L IS LESS THAN 71  GO TO E.
         00410        MOVE 0 TO L.
         00411        IF M = NUMBER, GO TO SLUTT.
         00415        GO TO RETUR.
         00420    ERROR.
         00425        STOP RUN.
         00430    SLUTT.
                      MOVE 0 TO M.
         00435        CLOSE INNFLYTT, FORBEST.
         00440        ALTER A TO PROCEED TO B.
         00445    RETUR.
         00450        ENTER LINKAGE.
         00455        RETURN.
         00460        ENTER COBOL.
         00465        NOTE VED LESNING AV INNFLYTTINGSANDELE OG FORELOEPIG BESTAND
         00470        FOR SISTE KOMMUNE, SPOLES BÅNDENE AUTOMATISK TILBAKE.
```

RES2

```
LINE NO. SEQ. NO.          SOURCE STATEMENT                                   CBD CL3-9

         00100    IDENTIFICATION DIVISION.
         00105    PROGRAM-ID. 'CORES2'.
         00110    REMARKS. SUBROUTINE SOM SKRIVER NY BESTAND PÅ TAPE OG DISK
         00115        OG INNFLYTTERE PÅ TAPE.
         00120    ENVIRONMENT DIVISION.
         00125    INPUT-OUTPUT SECTION.
         00130    FILE-CONTROL.
         00135        SELECT PROGRES ASSIGN TO 'SYS004' UTILITY 2400,
                          RESERVE NO ALTERNATE AREA.
         00140        SELECT BEVEGELSE ASSIGN TO 'SYS007' UTILITY 2400,
                          RESERVE NO ALTERNATE AREA.
         00145        SELECT BEDISK ASSIGN TO 'SYS001' UTILITY 2314,
                          RESERVE NO ALTERNATE AREA.
         00150    DATA DIVISION.
         00155    FILE SECTION.
         00160    FD  PROGRES, DATA RECORDS ARE PRORECOR1, LABEL RECORDS ARE
         00165        STANDARD, RECORDING MODE IS F.
         001651 01 PRORECOR1.
         001652       02 AR, PICTURE 99.
         001653       02 MUNNR, PICTURE IS 9(4).
                      02 FY1  PICTURE XX.
         001654       02 TOVE.
         001655       03 DON, OCCURS 100 TIMES.
         001656       04 DOL, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
         00166    FD  BEDISK, DATA RECORDS ARE PRORECORD, LABEL RECORDS ARE
         00167        STANDARD, RECORDING MODE IS F, BLOCK CONTAINS 3 RECORDS.
         00170    01  PRORECORD.
         00175        02 AAR, PICTURE IS 99.
         00180        02 KOMMNR, PICTURE IS 9(4).
                      02 FY2  PICTURE XX.
         00185        02 SKRUE.
         00190        03 DONALD, OCCURS 100 TIMES.
         00195        04 DOLLY, OCCURS 2 TIMES, USAGE IS COMPUTATIONAL-1.
         00200    FD  BEVEGELSE, DATA RECORDS ARE INNRECORD, LABEL RECORDS ARE
         00205        STANDARD, RECORDING MODE IS F.
         00210    01  INNRECORD.
         00215        02 PROGAR, PICTURE 99.
                      02 FY3  PICTURE XX.
         00220        02 KOMNR, PICTURE 9(4).
         00225        02 INNFLYTTERE.
```

```
LINE NO. SEQ. NO.          SOURCE STATEMENT

         00230       03 INNF, OCCURS 71  TIMES.
         00235       04 INN, OCCURS 2 TIMES, PICTURE S9999.
         00240  WORKING-STORAGE SECTION.
         00245  77  L, PICTURE 9(5), VALUE IS 0.
         00250  77  K, PICTURE 9(5), VALUE IS 0.
         00255  LINKAGE SECTION.
         00260  77  SWITCH, PICTURE 9(5), USAGE IS COMPUTATIONAL.
         00265  77  J, PICTURE 9(5), USAGE IS COMPUTATIONAL.
         00270  77  NR, PICTURE 9(5), USAGE IS COMPUTATIONAL.
         00275  01  BA.
         00280       02 BAB, OCCURS 100 TIMES.
         00285       03 BN, OCCURS 2 TIMES,
         00290       USAGE IS COMPUTATIONAL-1.
         00295  01  IA.
         00300       02 IB, OCCURS 71  TIMES.
         00305       03 INF, OCCURS 2 TIMES,
         00310       USAGE IS COMPUTATIONAL-1.
         00315  PROCEDURE DIVISION.
         00320       ENTER LINKAGE.
         00325       ENTRY 'RES2', USING NR, BN (1, 1), INF (1, 1), SWITCH, J.
         00330       ENTER COBOL.
         00335  RES2.
         00340  AA. GO TO AB.
         00345  AB. OPEN OUTPUT BEVEGELSE, PROGRES.
         00350  AG. OPEN OUTPUT BEDISK.
         00355       ALTER AA TO PROCEED TO AC.
         00360  AC. IF SWITCH IS EQUAL TO 1 GO TO SLUTT,
         00365       ELSE IF SWITCH IS EQUAL TO 0 GO TO SLUTT1.
         00370  AD. MOVE NR TO KOMMNR.
         00375       MOVE NR TO KOMNR.
         00376       MOVE NR TO MUNNR.
         00380       MOVE J TO AAR.
         00385       MOVE J TO PROGAR.
         00386       MOVE J TO AR.
         00390  AE. ADD 1 TO K.
         00395  AF. ADD 1 TO L.
         00400       MOVE BN (K, L) TO DOLLY (K, L).
         00401       MOVE BN (K, L) TO DOL (K, L).
         00410       IF L IS LESS THAN 2 GO TO AF.
         00415       MOVE 0 TO L.
         00420       IF K IS LESS THAN 100 GO TO AE.
         00425       MOVE 0 TO K.
                AE2.
                     ADD 1 TO K.
                AF2.
                     ADD 1 TO L.
  S      00405       MULTIPLY INF (K, L) BY 1 GIVING INN (K, L) ROUNDED.
                     IF L IS LESS THAN 2 GO TO AF2.
                     MOVE 0 TO L.
                     IF K IS LESS THAN 71 GO TO AE2.
                     MOVE 0 TO K.
                     MOVE ZEROES TO FY1, FY2, FY3.
         00440       WRITE PRORECORD.
         00441       WRITE PRORECOR1.
         00445       WRITE INNRECORD.
         00450       GO TO RETUR.
         00455  SLUTT.
         00460       CLOSE BEDISK, PROGRES, BEVEGELSE.
         00461       GO TO RETUR.
         00465  SLUTT1.
         00475       CLOSE BEDISK.
         00480       ALTER AA TO PROCEED TO AG.
         00485  RETUR.
         00490       ENTER LINKAGE.
         00495       RETURN.
         00496       ENTER COBOL.
         00497       NOTE TILBAKESPOLING AV BÅND OG NULLSTILLING AV DISK ER TATT
         00498       HENSYN TIL I PROGRAMMET.
```

SHØ/WA, 5/1-73

The Programme For Optimum Classification

by Synnøve Høvset

C o n t e n t s : Page

1. Introduction

Methods for splitting data into homogeneous groups are given by E. Gilje and I. Thomsen in [1]. The computer programme, appendix 1, is an example of the use of method II described in [1].

The computer programme consists of a main programme, a subroutine and a subfunction.

2. Comments to the computer programme

Main programme (FORTRAN)

The values for the parameters NR and K1 are read from cards, line 4 and 9 in the programme.

Nr = no. of municipalities.

K1 = no. of groups.

Data are read from inputtape, and placed into a 2-dim.-array, DATA (I,L). A record contains 2 numbers with format F9.7 and F5.2 for each municipality. These two numbers are written in DATA (I,1) and DATA (I,2), respectively. The numbers in DATA (I,2) are not used, refer to line 16, where J in DATA (I,J) is set equal to 1. The number in DATA (I,1) is the gross reproduction rate for one municipality, I = 1, 2, ..., NR.

The formula in line 15 corresponds to formula 4.1 in [1].

Subroutine SIMOPT (FORTRAN)

Subroutine to minimise or maximise a function with maximum 20 variables. Entry point for the routine is SIMOPT, and it requires 3 468 bytes.

Use: CALL SIMOPT (SUMSQ, N, P, DELTA, PMIN).

Parameterdescription:

SUMSQ - The function to be minimised.

N - No. of variables in SUMSQ.

P - 2-dimensional array with 21 rows and N columns.

The N+1 first rows represent N+1 points in the P-room (a simplex). At CALL, the starting values for the search must be in row 1. At RETURN, the first row contains the variable values for the minimum found.

DELTA - Parameter to decide the starting simplex. If DELTA = 0.0, the user has to define the whole starting simplex himself. If DELTA is not equal to 0, SIMOPT will make a regular simplex in P, by adding DELTA in turn into the different variables. The search starts with the chosen simplex according with DELTA.

FMIN - The minimum found.

Necessary subroutines and subfunctions:

SUMSQ (VAR) - VAR is a one-dimensional array which contains the variables.

Method: SIMOPT uses Nelder's and Mead's simplex-method. (Nelder, J.A. and Mead, R. (1965): "A simplex method for function minimization". The computer Journal, 7 (4), 308-313.)

Remarks:

SUMSQ has to be EXTERNAL defined in the main programme.

SIMOPT will find local minimum, but not necessarily absolute minimum. If the result seems unreasonable, or the programme is canceled for no obvious reason, try other start conditions.

If the original simplex lies in a sub room of the whole P-room, the search will take place in the sub room. The search will be limited to a certain area, by defining great function values outside the area.

SIMOPT has the best working conditions when the variable area for all variables are the same order of quantity. This may, if necessary, be done by scaling in the subfunction SUMSQ.

N must be greater than 0, less than 21. If N is greater than 20, SIMOPT will write an error message and terminate the job.

Subfunction SUMSQ (FORTRAN)

The function used, corresponds to formula 2.1 in 1 .

Reference

[1] Gilje, Eivind and Ib Thomsen (1970): "Two methods for splitting data into homogeneous groups", Statistical Review of the Swedish National Central Bureau of Statistics, 1970, No. 4.

Appendix 1

MAIN PROGRAMME

```
DOS FORTRAN IV 360N-FO-479 3-1

LINE                    MAINPGM           DATE       24/01/70

      DIMENSION P(21,20),IVAR(20)
      EXTERNAL SUMSQ
      COMMON DATA(500,2),K,J,NR,GJ(20)
      READ (1,110) NR
  110 FORMAT (I4)
      DO 7 I=1,NR
    7 READ (14,2) (DATA(I,L),L=1,2)
    2 FORMAT (11X,F9.7,1X,F5.2,6X)
    1 READ (1,100,END=50) K1
  100 FORMAT (I2)
      N=K1-1
      K=N
      DELTA=10.
      DO 3 I=1,N
    3 P(1,I)=NR/K1-   DELTA/2.
      DO 4 J=1,1
      CALL SIMOPT (SUMSQ,N,P,DELTA,FMIN)
      DO 5 I=1,N
    5 IVAR(I)=P(1,I)+0.5
      IANT=0
      DO 6 I=1,N
    6 IANT=IANT+IVAR(I)
      IVAR(K1)=NR-IANT
      WRITE (8,102) K1, (IVAR(I),GJ(I),I=1,KL)
  102 FORMAT (1X,I2/(1X,I4,F6.2))
    4 WRITE (8,103) FMIN
  103 FORMAT (1X,F13.5)
      GO IO 1
   50 CONTINUE
      END
```

SUBFUNCTION SUMSQ

```
DOS FORTRAN IV 360N-FO-479 3-1

LINE                                 SUMSQ          DATE     24/01/70

      FUNCTION SUMSQ(VAR)
      DIMENSION VAR(20),IVAR(20),SIGMA(20)
      COMMON DATA(500,2),K,J,NR,GJ(20)
      DO 1 I=1,K
    1 IVAR(I)=VAR(I)+0.5
      IANT=0
      DO 6 I=1,K
    6 IANT=IANT+IVAR(I)
      IVAR(K+1)=NR-IANT
      ITEL=0
      K=K+1
      II=0
      DO 2 I=1,K
      C=0.
      D=0.
      ITEL=II+1
      II=IVAR(I)+ITEL-1
      DO 3 M=ITEL,II
    3 C=C+DATA(M,J)
      C=C/IVAR(I)
      GJ(I)=C
      DO 4 L=ITEL,II
      H=(DATA(L,J)-C)**2
    4 D=D+H
    2 SIGMA(I)=D
      C=0.
      DO 5 I=1,K
    5 C=C+SIGMA(I)
      SUMSQ=C
      K=K-1
      RETURN
      END
```

SUBROUTINE SIMOPT

```
DOS FORTRAN IV 360N-FO-479 3-6          SIMOPT              DATE    05/01/73
                SUBROUTINE SIMOPT(FUNC,N,P,DELTA,FMIN)
                DIMENSION P(21,1),D(20),PBAR(20),PS(20),PSS(20),Y(21)
                IF(N-20)37,37,85
             85 WRITE (8,1) N
              1 FORMAT(23H TERMINERT AV SIMOPT,N=,I6,12H ER FOR STOR)
                STOP
      C         INITIALIZATION
             37 ALPHA = 1.
                BETA = 0.5
                GAMMA = 2.
                YMINX=1.E+75
                N1=N+1
                AN = N
                AN1 = N1
                ITP = 3*N1
                IF(ITP-15)50,51,51
             50 ITP = 15
             51 IF(ITP-30)52,52,70
             70 ITP = 30
             52 IFSTR=0
                A=0.
                B=0.
                C=0.
                IF(DELTA)71,10,71
             71 DO 11 I=2,N1
                DO 12 J=1,N
             12 P(I,J)=P(1,J)
             11 P(I,I-1) = P(I,I-1)+DELTA
             10 DO 13 I=1,N1
                DO 14 J=1,N
             14 D(J)=P(I,J)
             13 Y(I) = FUNC(D)
             60 NC=0
                MC=0
             15 K=1
                L=1
                DO 16 I=2,N1
                IF(Y(I)-Y(K))53,53,72
             72 K = I
             53 IF(Y(I)-Y(L))93,16,16
             93 L = I
             16 CONTINUE
                DO 17 J=1,N
                PBAR(J)=0.
                DO 18 I=1,N1
             18 PBAR(J)=PBAR(J)+P(I,J)
             17 PBAR(J)=(PBAR(J)-P(K,J))/AN
                DO 19 J=1,N
             19 PS(J)=(1.+ALPHA)*PBAR(J)-ALPHA*P(K,J)
                YS = FUNC(PS)
                IF(YS-Y(L))73,20,20
      C         YS IS NEW MINIMUM
```

```
      73 DO 21 J=1,N
      21 PSS(J)=(1.+GAMMA)*PS(J)-GAMMA*PBAR(J)
         YSS = FUNC(PSS)
         IF(YSS-YS)23,22,22
C        YSS IS NEW MINIMUM. SUCCESSFUL EXPANSION
      23 DO 24 J=1,N
      24 P(K,J)=PSS(J)
         Y(K)=YSS
         GO TO 25
C        REFLECTED POINT IS NOT NEW MINIMUM
      20 TMP=Y(K)
         Y(K)=Y(L)
         DO 26 I=1,N1
         IF(YS-Y(I))22,26,26
      26 CONTINUE
         GO TO 27
      22 DO 28 J=1,N
      28 P(K,J)=PS(J)
         Y(K)=YS
         GO TO 25
C        YS IS GREATER THAN SECOND GREATEST POINT OF SIMPLEX
      27 Y(K)=TMP
         IF(YS-Y(K))74,74,29
      74 DO 30 J=1,N
      30 P(K,J)=PS(J)
         Y(K)=YS
      29 DO 31 J=1,N
      31 PSS(J)=BETA*P(K,J)+(1.-BETA)*PBAR(J)
         YSS = FUNC(PSS)
         IF(YSS-Y(K))23,23,75
C        FAILED CONTRACTION. REDEFINE SIMPLEX
      75 DO 32 J=1,N
         DO 32 I=1,N1
      32 P(I,J)=.5*(P(I,J)+P(L,J))
      25 NC=NC+1
C        TEST FOR CONVERGENCE
         IF(IFSTR)61,61,76
      76 IF(NC-ITP)15,77,77
      77 IFSTR=0
         NC=0
      61 SY=0.
         SYY=0.
         DO 33 I=1,N1
      33 SY=SY+Y(I)
         SY=SY/AN1
         DO 34 I=1,N1
      34 SYY=SYY+(Y(I)-SY)**2
         A=B
         B=C
         C=SYY
         IF(NC-3)15,87,87
      87 IF(C-A)78,62,62
      78 MC=0
         GO TO 15
```

```
0100         62 IF(MC)79,79,63
0101         79 MC=1
0102            GO TO 15
0103         63 M=L
0104            IF(Y(K)-Y(L))80,64,64
0105         80 M=K
0106         64 YMIN=Y(M)
0107            SD=SQRT(C/AN1)
0108            IF(ABS(YMINX-YMIN)-.5*SD)65,65,81
0109         81 YMINX=YMIN
0110            IFSTR=1
0111            GO TO 60
           C    SET OPTIMUM POINT IN TOP ROW FOR OUTPUT
0112         65 IF(M-1)82,35,82
0113         82 DO 36 J=1,N
0114            TMP=P(1,J)
0115            P(1,J)=P(M,J)
0116         36 P(M,J)=TMP
0117            TMP=Y(1)
0118            Y(1)=Y(M)
0119            Y(M)=TMP
0120         35 FMIN=YMIN
0121            RETURN
0122            END
```

KEB/IVG, 2/11-72

SIMULTANE FORDELINGER AV HUSHOLDNINGENE I INNTEKTS/FORBRUKSUNDERSØKELSEN 1967 ETTER INNTEKT OG TOTAL FORBRUKSUTGIFT

Av Erik Biørn

I Skatteforskningsgruppen pågår et stadig arbeid med sikte på å utbygge og forbedre analysemodeller for personbeskatningen. Det er i den forbindelse behov for å øke vår innsikt i konsumadferden på mikronivå. Spesielt er det ønskelig å kartlegge mulighetene for å etablere empiriske relasjoner mellom inntekt og konsum på husholdningsnivå. Slik innsikt er også ønskelig for å kunne forbedre relasjonene for det private konsum i makro-økonomiske planleggings- og prognosemodeller (f.eks. MODIS).

Et sentralt problem i denne forbindelse består i å etablere et hensiktsmessig uttrykk for "konsummotiverende inntekt for en husholdning" - om da et slikt begrep har noen god mening. Om en ikke greier å fastlegge en hensiktsmessig definisjon basert på apriori overveielser, kunne en etablere forskjellige alternative inntektsdefinisjoner og ved hjelp av regresjon prøve å finne ut hvilket inntektsbegrep som (sammen med andre variable) "forklarer" husholdningenes konsumutgift best. Dette er ingen ideell fremgangsmåte, med den kunne vel være brukbar som en første tilnærmelse.

Vedlagt følger to frekvenstabeller utarbeidet på grunnlag av materialet fra inntekts/forbruksundersøkelsen 1967, som jeg synes belyser viktige sider ved dette problemet. Den file som ligger til grunn, inneholder opplysninger om antatt inntekt ved kommune- og statsskatteligningen, pensjonsgivende inntekt i og utenfor tjenesteforhold samt forbruksoppgaver for 3 645 husholdninger. Disse har både ført månedsregnskap og gitt tilleggsopplysninger i et årsintervju. Oppgavene fra årsintervjuet er benyttet for de varegrupper hvor slike oppgaver er innhentet. (Filen ble laget spesielt for modellbyggingsformål ved utgangen av 1970.)

Følgende symboler er innført:

R^S = Antatt inntekt ved statsskatteligningen[1)]

R^K = " " " kommuneskatteligningen

R^P = Summen av pensjonsgivende inntekt i og utenfor tjenesteforhold

C = Total forbruksutgift inkl. utgifter til kjøp av egne transportmidler

C^* = Total forbruksutgift ekskl. utgifter til kjøp av egne transportmidler.

1) For de skattytere som betaler inntektsskatt til staten, 0 ellers.

Tabell 1 gir den simultane fordeling ~~av~~ R^S og C, mens tabell 2 gjelder den simultane fordeling av R^P og C. De tilsvarende simultane fordelinger av R^S og C^* og av R^P og C^* er gitt i tabellene 3 og 4. Det fremgår at over 1/3 av husholdningene hadde $R^S = 0$, og av disse fantes det husholdninger med C større enn 70 000 kroner (tilsvarende gjelder for C^*). Henimot 1/4 av husholdningene i utvalget hadde $R^P = 0$, mens 514 husholdninger eller 14% hadde $R^K = 0$. Det er ikke undersøkt hvordan disse husholdningene fordeler seg på sosialgrupper.

Tabellene viser også at en betydelig andel av husholdningene i utvalget figurerer med en total forbruksutgift som overstiger både antatt inntekt og pensjonsgivende inntekt. De nøyaktige tall, fordelt på sosialgrupper, er følgende:

Sosialgruppe	Antall husholdninger hvor				Totalt antall husholdninger
	$C>R^S$	$C^*>R^S$	$C>R^P$	$C^*>R^P$	
Selvst. prim.nær.	439	424	403	390	539
Selvst. ellers	167	151	152	137	311
Lønnstakere	1 108	1 007	921	812	2 173
Ikke yrkesaktive	533	533	612	612	622
Sum	2 247	2 115	2 088	1 951	3 645

De hovedspørsmål det etter dette er grunn til å stille seg, og som vi håper at fremtidige inntekts/forbruksundersøkelser kan bidra til å kaste lys over, er følgende (upresist formulert):[2)]

1. Hva livnærer de husholdninger seg av som opptrer med nullinntekt definert ut fra skatteformål?
2. Hva er grunnene til at over halvparten av husholdningene har et registrert konsum som er større enn "inntekt før skatt"?

Før vi blir i stand til å besvare disse spørsmål tilfredsstillende, tror jeg det vil være vanskelig å gjøre videre fremstøt når det gjelder empirisk analyse av husholdningenes konsum- og spareadferd.

2) Jfr. imidlertid: Statistikk over lavinntektsgrupper 1967 (NOS A 462).

Tabell 1. Inntekts/forbruksundersøkelsen 1967. Fordeling av husholdningene etter antatt inntekt ved statsskatteligningen (R^S) og total forbruksutgift inkl. kjøp av egne transportmidler (C).

R^S \ C	0	1-5000	5001-10000	10001-15000	15001-20000	20001-25000	25001-30000	30001-35000	35001-40000	40001-50000	50001-70000	70001-∞	Σ
0	0	88	285	276	207	157	107	66	41	28	16	4	1275
1-5000	0	0	2	5	9	3	5	0	1	1	1	1	28
5001-10000	0	6	25	26	18	10	4	5	1	3	2	0	100
10001-15000	0	4	34	67	34	26	11	7	7	3	3	1	197
15001-20000	0	2	43	92	98	65	28	22	20	16	4	1	391
20001-25000	0	1	35	83	158	120	75	55	25	33	7	1	593
25001-30000	0	0	10	32	93	111	65	46	34	34	22	1	448
30001-35000	0	0	6	20	37	67	50	25	24	30	9	1	269
35001-40000	0	0	3	7	17	21	26	24	5	18	11	1	133
40001-50000	0	0	2	4	10	9	23	15	12	17	9	2	103
50001-70000	0	0	0	1	5	6	11	14	7	16	14	2	76
70001-∞	0	0	0	1	1	1	2	2	1	12	9	3	32
Σ	0	101	445	614	687	596	407	281	178	211	107	18	3645

Tabell 2. Inntekts/forbruksundersøkelsen 1967. Fordeling av husholdningene etter pensjonsgivende inntekt (R^P) og total forbruksutgift inkl. kjøp av egne transportmidler (C).

R^P \ C	0	1-5000	5001-10000	10001-15000	15001-20000	20001-25000	25001-30000	30001-35000	35001-40000	40001-50000	50001-70000	70001-∞	Σ
0	0	86	273	196	115	65	35	26	14	10	5	1	826
1-5000	0	0	12	4	7	4	6	2	3	0	4	0	42
5001-10000	0	11	48	64	37	23	10	7	7	7	1	1	216
10001-15000	0	2	24	77	58	50	14	12	16	6	5	2	266
15001-20000	0	2	31	99	93	54	38	21	12	12	4	0	366
20001-25000	0	0	36	87	165	144	75	55	30	20	12	1	625
25001-30000	0	0	12	48	110	114	79	55	30	42	11	3	504
30001-35000	0	0	4	26	60	78	69	33	26	34	14	1	345
35001-40000	0	0	3	8	25	41	35	27	16	23	11	1	190
40001-50000	0	0	2	5	16	23	46	42	24	57	40	8	263
50001-70000	0	0	0	0	1	0	0	1	0	0	0	0	2
70001-∞	0	0	0	0	0	0	0	0	0	0	0	0	0
Σ	0	101	445	614	687	596	407	281	178	211	107	18	3645

Tabell 3. Inntekts/forbruksundersøkelsen 1967. Fordeling av husholdningene etter antatt inntekt ved statsskatteligningen (R^S) og total forbruksutgift ekskl. kjøp av egne transportmidler (C^*).

R^S \ C^*	0	1-5000	5001-10000	10001-15000	15001-20000	20001-25000	25001-30000	30001-35000	35001-40000	40001-50000	50001-70000	70001-∞	Σ
0	0	88	288	290	216	153	114	53	37	21	14	1	1275
1-5000	0	0	2	5	10	2	5	1	1	2	0	0	28
5001-10000	0	6	26	29	18	8	3	5	1	3	1	0	100
10001-15000	0	4	36	68	36	26	12	6	6	1	1	1	197
15001-20000	0	2	48	101	106	62	23	20	17	9	3	0	391
20001-25000	0	1	36	98	168	122	86	39	20	20	3	0	593
25001-30000	0	0	11	40	105	117	68	56	22	15	13	1	448
30001-35000	0	0	6	27	47	69	51	28	16	19	5	1	269
35001-40000	0	0	3	7	19	26	30	21	9	10	8	0	133
40001-50000	0	0	2	5	10	11	28	14	11	15	6	1	103
50001-70000	0	0	0	1	5	7	11	17	9	18	7	1	76
70001-∞	0	0	0	1	1	1	2	2	2	14	8	1	32
Σ	0	101	458	672	741	604	433	262	151	147	69	7	3645

Tabell 4. Inntekts/forbruksundersøkelsen 1967. Fordeling av husholdningene etter pensjonsgivende inntekt (R^P) og total forbruksutgift ekskl. kjøp av egne transportmidler (C^*).

R^P \ C^*	0	1-5000	5001-10000	10001-15000	15001-20000	20001-25000	25001-30000	30001-35000	35001-40000	40001-50000	50001-70000	70001-∞	Σ
0	0	86	274	201	119	66	40	17	9	10	4	0	826
1-5000	0	0	12	4	8	3	7	3	3	1	1	0	42
5001-10000	0	11	48	68	38	21	10	9	7	2	2	0	216
10001-15000	0	2	27	80	62	44	18	12	12	5	3	1	266
15001-20000	0	2	36	105	92	54	37	20	10	8	2	0	366
20001-25000	0	0	40	107	177	142	76	34	27	15	6	1	625
25001-30000	0	0	12	61	121	117	87	51	25	21	8	1	504
30001-35000	0	0	4	28	75	86	67	42	18	17	8	0	345
35001-40000	0	0	3	12	31	44	39	28	12	13	7	1	190
40001-50000	0	0	2	6	17	27	52	45	28	55	28	3	263
50001-70000	0	0	0	0	1	0	0	1	0	0	0	0	2
70001-∞	0	0	0	0	0	0	0	0	0	0	0	0	0
Σ	0	101	458	672	741	604	433	262	151	147	69	7	3645

JMH/KB, 22/2-73

Moment estimators for the Hadwiger function

by Jan M. Hoem.

Contents

1. Introduction

A number of parametric functions have been suggested for the analytic graduation of human age-specific fertility rates. (For a review, see Hoem, 1972, Section 1.2.) One of these, given below, is usually called the Hadwiger function, since its use in fertility graduation was first suggested by Hadwiger (1940).

Studies of the empirical fitting properties of the Hadwiger function have been made on some occasions, and in particular by Yntema and by Gilje. (See Gilje and Yntema, 1971, and their references.) Various procedures for the estimation of the parameters of the function have been suggested, and the statistical properties of one set of such estimators, due to Yntema, have been discussed by Hoem (1972, Section 7.4).

For some reason, the standard moment estimators of these parameters do not seem to have appeared in the literature, although their derivation is quite straightforward. In connection with an extensive investigation of regional fertility in Norway currently being made in the Central Bureau of Statistics, it has been considered useful to see how the moment estimators compare to other suggestions. I have been asked to write up an account of how they are derived, in a form which will permit the reader to follow the mathematical argument step by step. The result is presented in this note.

The statistical properties of the estimators depend on those of the fertility rates which are graduated. They can be studied by the methods given by Hoem (1972). The derivation of such properties can be a useful exercise for students of this material, and it will not be given here.

In writing this note, incidentally, I discovered that there is an error in a formula for the cumulants of the Hadwiger function which I have given previously (Hoem, 1972, (7. 24).) The correct formula is presented in (5) below.

2. The Hadwiger density and its cumulants.

Let

(1) $$\phi(x) = H \left(\frac{T}{\pi}\right)^{1/2} x^{-3/2} \exp\left\{-H^2\left(\frac{T}{x} + \frac{x}{T} - 2\right)\right\} \text{ for } x > 0,$$

and let

(2) $$\psi(\lambda) = \ln \int_0^\infty e^{\lambda x} \phi(x)dx.$$

We first prove that

(3) $\psi(\lambda) = 2\,H^2\,\{1 - (1 - \lambda T/H^2)^{1/2}\}.$

Proof: Introduce $y = x^{-1/2}$ in (2). Then

$$\psi(\lambda) = \ln\,\{2\,H\,(\tfrac{T}{\pi})^{\frac{1}{2}}\,e^{\,2H^2}\int_0^\infty \exp\left[y^{-2}\,(\lambda - \tfrac{H^2}{T}) - H^2Ty^2\right]dy\}.$$

To proceed, we shall make use of the standard integral (see, e.g., Keyfitz, 1968, p. 150)

$$\int_0^\infty \exp\,[-(\frac{\alpha^2}{y^2} + \beta^2y^2)]\,dy = \frac{\sqrt{\pi}}{2\beta}\,e^{-2\alpha\beta}.$$

This gives

$$\psi(\lambda) = \ln\,\{2H(\tfrac{T}{\pi})^{\frac{1}{2}}\,e^{\,2H^2}\,\frac{\sqrt{\pi}}{2H\sqrt{T}}\,\exp\left[-2\,(H^2T(\tfrac{H^2}{T} - \lambda))^{\frac{1}{2}}\right]\},$$

from which (3) immediately follows. □

Formula (3) shows that $\psi(0) = 0$, which proves that ϕ is a probability density over $[0,\infty]$. We shall call it the Hadwiger density.

Let Y be a random variable with density ϕ, and let η_n be its n-th cumulant. Then

$$EY = \eta_1,\ \text{var}\ Y = \eta_2,\ E(Y-EY)^3 = \eta_3,$$

and so on. The function ψ is the cumulant generating function of ϕ, so that

$$\psi(\lambda) = \sum_{n=1}^\infty \frac{\eta_n}{n!}\,\lambda^n.$$

We shall prove that

(4) $\eta_1 = T,\quad \eta_2 = \frac{1}{2}\frac{T^2}{H^2},\quad \eta_3 = \frac{3}{4}\frac{T^3}{H^4},$

and

(5) $\eta_n = (2n-3)\,(2n-5)\cdots 3\cdot 1\cdot T^n/(2H^2)^{n-1}$ for $n \geq 2$.

Proof: Using the binomial expansion of $(1-x)^{\frac{1}{2}}$ in (3) and rearranging, we get

$$\psi(\lambda) = -2H^2 \sum_{n=1}^{\infty} \binom{1/2}{n} \left(-\frac{T}{H^2}\right)^n \lambda^n.$$

Thus $\eta_1 = T$ is immediate. For $n \geq 2$,

$$\binom{1/2}{n} = \frac{1}{2}\left(\frac{1}{2} - 1\right)\left(\frac{1}{2} - 2\right) \cdots \left(\frac{1}{2} - n + 1\right)/n!$$

$$= (-1)^{n-1}\left(n - \frac{3}{2}\right)\left(n - \frac{5}{2}\right) \cdots \frac{3}{2} \cdot \frac{1}{2} \cdot \frac{1}{2}/n!$$

$$= (-1)^{n-1}(2n - 3)(2n - 5) \cdots 3 \cdot 1 \cdot 2^{-n}/n!,$$

which gives (5). The formulas for η_2 and η_3 follow from (5). □

3. The shifted Hadwiger function.

Let

(6) $$h(x) = R\phi(x - d) \qquad \text{for } x > d,$$

$$R'_n = \int_d^{\infty} x^n \, h(x)dx \qquad \text{for } n \geq 0,$$

$$\mu_n = \int_d^{\infty} (x - d - T)^n \, h(x) \, dx \qquad \text{for } n \geq 2.$$

Then $R'_0 = R$ and $R'_1 = R(T + d)$, while

$$\mu_n = R \int_0^{\infty} (y - \eta_1)^n \, \phi(y)dy.$$

Thus, μ_n equals R times the n-th central moment of ϕ. By the results of Section 2 above, therefore,

$$\mu_2 = R\eta_2 = \frac{RT^2}{2H^2}$$

and

$$\mu_3 = R\eta_3 = \frac{3RT^3}{4H^4}.$$

Following Gilje and Yntema (1971), we shall call h the <u>shifted Hadwiger fertility function</u>. We shall derive estimators for its four parameters, R, T, H, and d, by the method of moments.

Let the observed (empirical) fertility rates be $\{\hat{\lambda}_x : x = \alpha, \alpha+1, \ldots, \beta - 1\}$, for given α and β. Let $U = T+d$, and introduce

$$\hat{R} = \Sigma\hat{\lambda}_x,$$

$$\hat{R}_1' = \Sigma x\hat{\lambda}_x,$$

$$\hat{U} = \hat{R}_1' / \hat{R},$$

$$\hat{\mu}_2 = \Sigma (x - \hat{U})^2 \hat{\lambda}_x,$$

$$\hat{\mu}_3 = \Sigma (x - \hat{U})^3 \hat{\lambda}_x.$$

Then $\hat{R}$ is an estimator for R. Estimators $\hat{T}$, $\hat{H}$, and $\hat{d}$ for the three other parameters are obtained as the solution of the following equations:

(7) $$\hat{U} = \hat{T} + \hat{d},$$

(8) $$\hat{\mu}_2 = \frac{\hat{R}\hat{T}^2}{2\hat{H}^2},$$

(9) $$\hat{\mu}_3 = \frac{3\hat{R}\hat{T}^3}{4\hat{H}^4}.$$

Squaring (8), dividing the result by (9), and rearranging, we get

(10) $$\hat{T} = \frac{3\hat{\mu}_2^2}{\hat{R}\hat{\mu}_3}.$$

Formula (7) then gives $\hat{d}$ as $\hat{U} - \hat{T}$, and the combination of (8) and (10) gives

(11) $$\hat{H} = \frac{3\ \hat{\mu}_2^{3/2}}{(2\hat{R})^{1/2}\ \hat{\mu}_3}.$$

References.

Hoem, Jan M. (1972): "On the statistical theory of analytic graduation." Proc. Sixth Berkeley Symp. Math. Stat. Probab., **1**: 569 - 600. Reprinted as Article No. 49 from the Central Bureau of Statistics of Norway.

Gilje, Eivind and L. Yntema (1971): "The shifted Hadwiger fertility function." Skand. Aktuarietidskr. **54**: 4 - 13.

Hadwiger, H. (1940): "Eine analytische Reproduktionsfunktion für biologische Gesamtheiten." Skand. Aktuarietidskr. **23**: 101 - 123.

Keyfitz, Nathan (1968): "Introduction to the Mathematics of Population." Addison-Wesley.